

CLAIMS

What is claimed as new and desired to be protected by Letters Patent of the United States is:

1. A system for assigning a unique network identifier to each program invoked on a computer, the system comprising:
a computer obtaining a plurality of network identifiers, the computer comprising:
an interface mechanism selecting, from the plurality of network identifiers, a first network identifier for a first program invoked on the computer and selecting a second network identifier, different from the first network identifier, for a second program invoked on the computer, and associating the first network identifier with the first program and associating the second network identifier with the second program; and
a network communication interface, in communication with the interface mechanism, transmitting the first network identifier with a network communication of the first program, and transmitting the second network identifier with a network communication of the second program.

2. The system of claim 1, wherein the network identifier comprises an internet protocol address.
3. The system of claim 1, wherein the network identifier comprises a host name.
4. The system of claim 1, wherein one of the first program and the second program comprises a user session hosted by the computer.
5. The system of claim 1, wherein one of the first program and the second program comprises one of an application isolation environment and an application.
6. The system of claim 1, wherein the computer obtains at least one of the plurality of network identifiers from a server.
7. The system of claim 4, wherein the server comprises a Dynamic Host Configuration Protocol server.
8. The system of claim 1, wherein the computer obtains at least one of the plurality of network identifiers from a storage location.

9. The system of claim 1, wherein the computer obtains at least one of the plurality of network identifiers from a network identifier generator.
10. The system of claim 1, wherein the interface mechanism selects the first network identifier for the first program during an establishment of the first program.
11. The system of claim 1, wherein the interface mechanism selects the second network identifier for the second program during an establishment of the second program.
12. The system of claim 4, wherein the computer concurrently hosts a first user session and a second user session.
13. The system of claim 4, wherein the computer hosts a second user session subsequent to the hosting of a first user session.
14. The system of claim 1, wherein the interface mechanism provides the first network identifier of the first program in response to a name resolution request of the first program and provides the second network identifier of the second

program in response to a name resolution request of the second program.

15. The system of claim 1, wherein at least one of the plurality of network identifiers is allocated to a user of the computer.
16. The system of claim 1, wherein the interface mechanism comprises a first TCP stack.
17. The system of claim 16, wherein the interface mechanism comprises a socket library for communication with the network communication interface.
18. The system of claim 17, wherein the socket library comprises a WinSock application programming interface.
19. The system of claim 17, wherein the interface mechanism binds the first network identifier to the first program for socket communication with the network communication interface.
20. The system of claim 17, wherein the interface mechanism binds the second network identifier to the second program

for socket communication with the network communication interface.

21. The system of claim 16, wherein the interface mechanism comprises a second TCP stack.
22. The system of claim 1, wherein the interface mechanism comprises a network packet-manipulation filter.
23. A system for assigning a unique loopback address to each program invoked on a computer, the system comprising:
a computer obtaining a plurality of loopback addresses, the computer comprising:
an interface mechanism selecting, from the plurality of loopback addresses, a first loopback address for a first program invoked on the computer and selecting a second loopback address, different from the first loopback address, for a second program invoked on the computer, and associating the first loopback address as a local host address of the first program and associating the second loopback address as a local host address of the second program; and
a loopback interface, in communication with the interface

mechanism, transmitting the first loopback address with an inter-process communication of the first program, and transmitting the second loopback address with an inter-process communication of the second program.

24. The system of claim 23, wherein one of the first program and the second program comprises a user session hosted by the computer.
25. The system of claim 23, wherein one of the first program and the second program comprises one of an application isolation environment and an application.
26. The system of claim 23, wherein the interface mechanism selects the first loopback address for the first program during an establishment of the first program.
27. The system of claim 23, wherein the interface mechanism selects the second loopback address for the second program during an establishment of the second program.
28. The system of claim 23, wherein the computer obtains at least one of the plurality of loopback addresses from a server.

29. The system of claim 23, wherein the computer obtains at least one of the plurality of loopback addresses from a storage location.
30. The system of claim 23, wherein the system comprises a loopback address generator to generate at least one of the plurality of loopback addresses.
31. A method for assigning a unique network identifier to each program invoked by a computer, the method comprising the steps of :
 - (a) obtaining a plurality of network identifiers;
 - (b) selecting, from the plurality of network identifiers, a first network identifier for a first program invoked on a computer, and a second network identifier, different from the first network identifier, for a second program invoked on the computer;
 - (c) associating the first network identifier with network communication of the first program and associating the second network identifier with network communication of the second program; and
 - (d) transmitting the first network identifier with a

network communication of the first program, and
transmitting the second network identifier with a network
communication of the second program.

32. The method of claim 31, wherein the network identifier comprises an internet protocol address.
33. The method of claim 31, wherein the network identifier comprises a host name.
34. The method of claim 31, wherein one of the first program and the second program comprises a user session hosted by the computer.
35. The method of claim 31, wherein one of the first program and the second program comprises one of an application isolation environment and an application.
36. The method of claim 31, wherein step (a) further comprises obtaining, from a server, at least one of the plurality of network identifiers.

37. The method of claim 31, wherein step (a) further comprises obtaining, from a Dynamic Host Configuration Protocol server, at least one of the plurality of network identifiers.
38. The method of claim 31, wherein step (a) further comprises obtaining, by the computer, at least one of the plurality of network identifiers from a storage location.
39. The method of claim 31, wherein step (a) further comprises obtaining, by the computer, at least one of the plurality of network identifiers from a network identifier generator.
40. The method of claim 31, wherein step (b) further comprises selecting the first network identifier for the first program during an establishment of the first program.
41. The method of claim 31, wherein step (b) further comprises selecting the second network identifier for the second program during an establishment of the second program.
42. The method of claim 34, wherein step (b) further comprises hosting concurrently, by the computer, a first user session and a second user session.

43. The method of claim 34, wherein step (b) further comprises hosting, by the computer, a second user session subsequent to the hosting of a first user session.
44. The method of claim 31, wherein step (d) further comprises providing the first network identifier of the first program in response to a name resolution request of the first program and the second network identifier of the second program in response to a name resolution request of the second program.
45. The method of claim 31, wherein step (a) further comprises allocating at least one of the plurality of network identifiers to a user of the computer.
46. The method of claim 31, wherein the method further comprises using a first TCP stack for network communication.
47. The method of claim 31, wherein the method further comprises interfacing with a network communication interface using a socket library.

48. The method of claim 31, wherein the method further comprises interfacing with a network communication interface using a WinSock application programming interface.
49. The method of claim 47, wherein the method further comprises binding the first network identifier to the first program for network communications using the socket library.
50. The method of claim 47, wherein the method further comprises binding the second network identifier to the second program for network communications using the socket library.
51. The method of claim 46, wherein the method further comprises a second TCP stack for network communication.
52. The method of claim 31, wherein the method further comprises interfacing with a network communication interface using a network packet-manipulation filter.
53. A method for assigning a unique loopback address to each program invoked on a computer, the method comprising

the steps of :

- obtaining a plurality of loopback addresses;
- selecting, from the plurality of loopback addresses, a first loopback address for a first program invoked on a computer, and a second loopback address different from the first loopback address, for a second program invoked on the computer;
- associating the first loopback address as a local host address of the first program and associating the second loopback address as a local host address of the second program; and
- transmitting the first loopback address with an inter-process communication of the first program, and
- transmitting the second loopback address with an inter-process communication of the second program.

54. The method of claim 53, wherein one of the first program and the second program comprises a user session hosted by the computer.

55. The method of claim 53, wherein one of the first program and the second program comprises one of an application isolation environment and an application.
56. The method of claim 53, wherein step (a) further comprises obtaining, from a Dynamic Host Configuration Protocol server, at least one of the plurality of loopback addresses.
57. The method of claim 53, wherein step (a) further comprises obtaining at least one of the plurality of loopback addresses from a storage location.
58. The method of claim 53, wherein step (a) further comprises obtaining at least one of the plurality of loopback addresses from a loopback address generator.
59. The method of claim 53, wherein step (b) further comprises selecting the first loopback address for the first program during an establishment of the first program.
60. The method of claim 53, wherein step (b) further comprises selecting the second loopback address for the second program during an establishment of the second program.

